# Game Development with Javascript

Overview & Tools

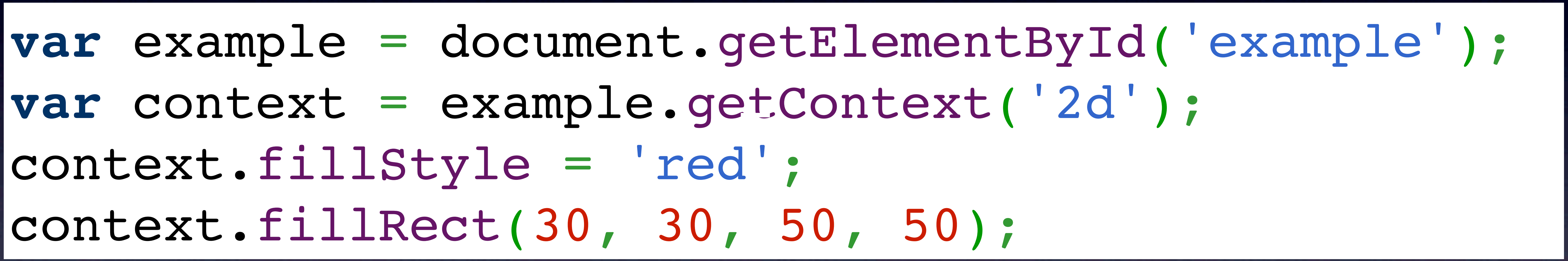
# Why Javascript?

- It’s everywhere and getting more popular.
- Growing set of libraries make it more attractive for a number of development tasks.
  - Client-Side Web - jQuery, Backbone.js
  - Server-Side Web - Node.js

# HTML 5 Canvas

- Draw directly to a segment of the browser without a plug-in.
- Utilize Javascript to provide the rendering code and game logic.
- Not currently as robust as Flash but on its way to being a rival.
- Supported in all current versions of major browsers.

# Canvas Drawing Example



```
var example = document.getElementById('example');
var context = example.getContext('2d');
context.fillStyle = 'red';
context.fillRect(30, 30, 50, 50);
```

Mozilla Canvas Tutorials

Source: http://en.wikipedia.org/wiki/Canvas_element

# Game Framework

- Provides abstractions for various game related tasks such as physics, animation, scene management, and audio.
- Leaves much of the game's tools and subsystem implementation to the developer.
- Popular frameworks - XNA, Cocos2D

# Game Engine

- Provides a cohesive end-to-end development solution including features such as physics, networking, rendering & lighting, editors, and asset pipeline.
- Handles the "hard stuff" so students can focus on making games and coding game logic.
- Popular engines - Unity 3D, Unreal (UDK), Source

# Javascript Game Dev. Frameworks & Engines

- There are a ton of them! Do some research or build your own!
- A random selection of Javascript engines/frameworks
  - Impact *(most robust as of this presentation date)*
  - MelonJS
  - Cocos2D Javascript

# Unity 3D

- Free* full-featured game engine.
- Export to Web, Windows, Mac OS X, iOS, Flash, Android, Linux.
- Gaining popularity with both independent and large developers.
- Used extensively in education.

* Free for companies with revenue under $100,000. iOS/Android versions cost $400 each. Pro versions are more.

# Why Unity

- Cohesive - No need to handle a large number of Javascript dependencies and possible browser incompatibilities. Single installer.
- Community - Large number of online resources
- Cost - Free for students.

# Unity 3D Scripting

- No access to underlying source code without appropriate license.
- Engine is completely scriptable through Javascript, C#, and Boo all thanks to the Mono project. The editor is also scriptable.
- Uses the syntax of the language but with libraries provided by Unity.

# Basic Concepts

- Everything in the game is a GameObject.
- GameObjects can be extended with Game Components.
- Game Components can be scripts.
- Game Objects and Components can be manipulated by scripts.
- Scripts can be partially exposed to the editor.

# Pedagogy

- Javascript is a language students can use in multiple contexts.
- Provide students with a way to visualize the product of their code.
- Provide students with a visual way to interact with their code.
- Unity's Game Object structure enforces the general ideas of OOP.
- My Experience - Student's learn mathematical concepts better when they can directly manipulate and visualize those concepts in-game.

# Demo

- Build a basic scene.
  - Explore the Unity interface.
  - Create game objects.
  - Create and attach scripts to game objects.
  - Interact with the game world.

# Links

- Unity3D
- Differences between Javascript and Unity Javascript

# Downloads & Contact

- jwertz@mc3.edu
- github.com/megawertz/Presentations